

INSTRUCCIONES

- Los valores nulos en las expresiones siempre dan como resultado un valor nulo

Instrucción	Definición	Ejemplos
. ED	Invoca al Editor del Sistema.	
. CLEAR SCR	Borra la pantalla	
. START fichero	Ejecuta el contenido almacenado en <i>fichero.sql</i> . Hay que especificar la ruta completa	
. RUN	Repite la ejecución de la última sentencia o de lo que hay en el buffer	. RUN . R
. LIST	Visualiza el contenido del buffer ó el contenido de la linea n	. LIST . LIST n . L n
. SAVE fichero	Almacena el contenido del buffer en <i>fichero.sql</i>	
. GET fichero	Recupera al buffer el contenido del fichero <i>fichero.sql</i>	
. INPUT	Añade una línea a continuación de la actual activa	
. DEL	Elimina la línea actual	
. SPOOL fichero	Todas las salidas por pantalla se almacenan en <i>fichero.lst</i>	
. SPOOL OFF	Libera el almacenamiento de salidas por pantalla	

Sentencia de creación de tablas y vistas

<pre>. CREATE TABLE tabla1 (col1 tipo_dato [NOT NULL], col2 tipo_dato [NOT NULL], ...);</pre>	<p>Crea la tabla1 con los campos especificados como col1, col2..., siendo cada campo del tipo y extensión definidos con tipo_dato.</p>	
<pre>. DESC tabla1</pre>	<p>Muestra la descripción de la tabla1, listando los nombres de los campos, tipo y extensión.</p>	
<pre>. CREATE TABLE tabla1 col1 tipo_dato CONSTRAINT nombre_constraint restricciones, col2 tipo_dato [NOT NULL] CONSTRAINT nombre_constraint restricciones, ... ; . CREATE TABLE tabla1 (col1 tipo_dato, col2 tipo_dato, ..., CONSTRAINT nombre_constraint PRIMARY KEY (col1, col2...), CONSTRAINT nombre_constraint FOREIGN KEY (col1,col2...) REFERENCES tabla2 (col1, col2...) [ON DELETE CASCADE], CONSTRAINT nombre_constraint CHECK cond ...);</pre>	<p>. Creación de una tabla con restricciones de columna.</p> <p>. Restricciones: . NOT NULL . UNIQUE . PRIMARY KEY . DEFAULT VALUE . REFERENCES tabla2 (col1, ...) [ON DELETE CASCADE] . CHECK cond</p> <p>En este otro caso las restricciones se ponen al final.</p>	

. CREATE TABLE tabla1 (col1, col2,...) AS consulta ;	Creación de una tabla a partir de los datos recogidos mediante una consulta en otra tabla.	
. DROP TABLE [usuario.]tabla1 [CASCADE CONSTRAINTS] ;	Se borra la tabla1 del usuario especificado [con restricciones incluidas]	
. ALTER TABLE tabla1 [ADD (col1 tipo_dato, col2 tipo_dato)] [MODIFY (col1 tipo_dato, col2 tipo_dato)] [ADD CONSTRAINT restricción] [DROP CONSTRAINT restricción] ;	Modifica la tabla, añadiendo nuevos campos, cambiando características de los campos, añadiendo restricciones, suprimiendo restricciones	
. CREATE VIEW vista1 [(col1, col2,...)] AS consulta ;	Se crea una vista con unos campos que se llamarán col1, col2... que se rellenan con los datos provenientes de una consulta	
. DROP VIEW vista1 ;	Se elimina la vista1	
. CREATE [PUBLIC] SYNONIM sinonimo1 FOR [usuario.]tabla1 ;	Crea un sinónimo para una tabla	
. DROP SYNONIM usuario.sinonimo1 ;	Se borra el sinonimo1	
. RENAME nombreaanterior TO nombrenuevo ;	Se renombra una tabla con un nombre nuevo	

Sentencia de consulta de datos

<p>. SELECT [ALL DISTINCT] [campo1, campo2,... *] FROM [tabla1 alias, tabla2 alias, ...] WHERE cond</p> <p>GROUP BY exp, exp</p> <p>HAVING cond</p> <p>ORDER BY campo1 [DESC ASC], campo2 [DESC ASC], ...</p>	<p>selecciona campos;</p> <p>de la(s) tabla(s); selecciona filas;</p> <p>agrupa las filas (los campos deben estar en la SELECT); filtra los grupos seleccionando y eliminando; clasifica la salida. Ordena los grupos.</p>	<p>. Se usan comillas dobles para poner cabecera a cada columna en el resultado: . SELECT campo1 "cabecera1" FROM tabla1</p>
	<p>. condiciones:</p> <p>where NOTA = 5 where EDAD < 26 ; EDAD > 26 where PRECIO >= 50 ; PRECIO <= 50 where DESC <> 15 ; DESC != 15 where a=8 AND b=3 where a=8 OR b=3 where NOT a=8 where APE LIKE 'A%' where GRUPO LIKE 'A_' where GRUPO IS NULL where GRUPO IS NOT NULL where DEPT IN (20,30,40) where DEPT BETWEEN 10 AND 50</p>	<p>. NOTAS:</p> <p>. Los literales van entre comillas simples</p> <p>. Si se usan comodines en una comparación (% , _) hay que utilizar LIKE y no =</p> <p>. Si el campo a comparar es de tipo CHAR, se puede utilizar: APE ='expr' ó APE LIKE 'expr'</p> <div><p>Se rellena a espacios toda la longitud de la variable CHAR</p></div>

Funciones de selección de datos		
OUTER JOIN (+)	Selecciona algunas filas de una tabla aunque no tengan correspondencia con las filas de la otra tabla. Se pone + a la tabla donde están las filas que no tendrán correspondencia en la otra	. SELECT A.DATO1, B.DATO2 FROM A B WHERE A.CLAVE = B.CLAVE(+) (en este caso habrá filas de la tabla A que no tengan correspondencia en la tabla B, por lo que los campos de la tabla B se rellenarán como NULL)
. UNION (ALL)	Une los resultados de dos consultas. Las filas duplicadas se reducen a una excepto si se indica ALL	. SELECT campo1, campo2 FROM tabla1 UNION SELECT campo1, campo2 FROM tabla2
. INTERSECT	Devuelve las filas comunes de dos consultas	. SELECT campo1, campo2 FROM tabla1 INTERSECT SELECT campo1, campo2 FROM tabla2
. MINUS	Devuelve las filas de tabla1 que no están en tabla2	. SELECT campo1, campo2 FROM tabla1 MINUS SELECT campo1, campo2 FROM tabla2

Funciones de manipulación de datos		
. INSERT INTO tabla (col1, col2,...) VALUES (val1, val2,...)	Inserta en las columnas los valores especificados. Si no se indican las col se insertarán valores en todos los campos	. INSERT INTO EMPLE NOM, TEL VALUES 'Jorge', '945566778' ;
. INSERT INTO tabla1 (col1, col2,...) SELECT (kol1, kol2,...) FROM tabla2 (WHERE cond) ;	Inserta en las col de la tabla1 los valores de las kol de la tabla2 que cumplan la condición	. INSERT INTO EMPLE (NOM, TEL) SELECT (NOM1, TEL1) FROM EMPLE2 WHERE POB='Bilbao';
. UPDATE tabla SET col1=val1, col2=val2 WHERE cond ;	Modifica las columnas de la tabla con los valores indicados en las filas que cumplan la condición	. UPDATE EMPLE SET PROV='Bizkaia' WHERE PROV='Vizcaya' ;

. UPDATE tabla SET col1=val1, col2=val2 WHERE col3 = (SELECT...);	Modifica las columnas de la tabla con los valores indicados en las filas que cumplan la condición fruto de la SELECT	. UPDATE EMPLE SET PROV='Bizkaia' WHERE PROV=(SELECT PROV FROM PROVINCIAS WHERE CP=48);
. UPDATE tabla1 SET (col, col2,...) = (select kol1,kol2... FROM tabla2) WHERE cond ;	Modifica las columnas de la tabla1 con los valores seleccionados de la tabla2, en las filas que cumplan la condición	. UPDATE EMPLE SET (APENOM) = (SELECT UPPER(APENOM) FROM EMPLE WHERE DNI=30456546) WHERE PROV='Bizkaia' ;
. DELETE (FROM) tabla WHERE cond ;	Borra de la tabla las filas que cumplan la condición	. DELETE FROM EMPLE WHERE PROV IN (SELECT PROV FROM DEPART) ;
. COMMIT ;	Valida todos los cambios hechas en la BD desde que abrimos la última sesión o desde el último COMMIT	
. ROLLBACK ;	Deshace los cambios en la BD desde el último COMMIT	
. AUTOCOMMIT	Es un COMMIT automático que se ejecuta tras cada sentencia	Para activar AUTOCOMMIT: (SET AUTOCOMMIT ON)
Otras funciones		
. DECODE (var,val1,cod1, val2, cod2..., valdefecto)	Si 'var' es igual a algún 'val', lo sustituye por su 'cod', y si no coincide, por el valor por defecto	. DECODE (SEXO,'M', 'MUJER','H', 'HOMBRE','SINDEFINIR')
. VSIZE (expresión)	Devuelve el número de bytes que ocupa exp	. SELECT VSIZE ('AMOREBIETA') FROM DUAL 10
. USER	Muestra el usuario que está conectado	. SELECT USER FROM DUAL SCOTT
. SHOW USER	Muestra qué usuario somos	. SHOW USER el usuario es 'SCOTT'
. UID	Devuelve el identificador del usuari actual	
. CONNECT usuario/password	Conecta el usuario con su password a la BD	. CONNECT SCOTT/TIGER CONECTADO

Funciones aritméticas			
. ABS (n)	Devuelve el valor absoluto de "n"	. SELECT ABS(2) FROM DUAL . SELECT ABS(-2) FROM DUAL	2 2
. CEIL (n)	Obtiene el valor entero inmediatamente superior o igual a "n"	. SELECT CEIL(13.2) FROM DUAL . SELECT CEIL(13) FROM DUAL . SELECT CEIL(-13.2) FROM DUAL	14 13 -13
. FLOOR (n)	Obtiene el valor entero inmediatamente inferior o igual a "n"	. SELECT FLOOR(13.2) FROM DUAL . SELECT FLOOR(13) FROM DUAL . SELECT FLOOR(-13.2) FROM DUAL	13 13 -14
. MOD (m, n)	Devuelve el resto de dividir m/n	. SELECT MOD(11,4) FROM DUAL . SELECT MOD(11,0) FROM DUAL	3 11
. NVL (valor, expresión)	Si "valor" es NULL, lo sustituye por "expresión"; si no, devuelve "valor"	. SELECT SALARIO, COMISION, SALARIO + NVL(COMISION, 0) FROM EMPL;	
. POWER (m, n)	Devuelve m ⁿ	. SELECT POWER(2,3) FROM DUAL . SELECT POWER(3,2) FROM DUAL	8 9
. ROUND (numero [,n])	Redondea número con "n" decimales	. SELECT ROUND(1.56, 1) FROM DUAL . SELECT ROUND(1.56) FROM DUAL . SELECT ROUND(127.56, -1) FROM DUAL	1.6 2 130
. SIGN (valor)	Indica el signo de "valor"	. SELECT SIGN(8.2) FROM DUAL . SELECT SIGN(-8.2) FROM DUAL	1 -1
. SQRT (n)	Devuelve la raíz cuadrada de n	. SELECT SQRT(9) FROM DUAL	3
. TRUNC (numero [,m])	Trunca número con "n" decimales	. SELECT TRUNC(1.56, 1) FROM DUAL . SELECT TRUNC(1.56) FROM DUAL . SELECT TRUNC(127.56, -1) FROM DUAL	1.5 1 120
. VARIANCE ([DISTINCT] valor)	Devuelve la varianza de los valores	. SELECT VARIANCE(SALARIO) FROM EMPL	
. AVG (n)	Calcula el valor medio de n ignorando los valores nulos	. SELECT AVG(EDAD) FROM EMPL	

. COUNT (*) expresión)	Cuenta todas las filas (*) o las que no tienen valor nulo	. SELECT COUNT (*) FROM EMPLE 24 . SELECT COUNT (COMISION) FROM EMPLE 10 . SELECT COUNT (DISTINCT TEL) FROM EMPLE 16
. MAX (expresión)	Calcula el máximo valor de la expresión	. SELECT MAX (SUELDO) FROM EMPLE 2500
. MIN (expresión)	Calcula el mínimo valor de la expresión	. SELECT MIN (SUELDO) FROM EMPLE 600
. SUM (expresión)	Obtiene la suma de los valores de la expresión	. SELECT SUM (SUELDO) FROM EMPLE 130000
Funciones de listas		
. GREATEST (valor1, valor2...)	Obtiene el mayor valor de la lista	. SELECT GREATEST (NOTA1, NOTA2, NOTA3) FROM NOTAS
. LEAST (valor1, valor2...)	Obtiene el menor valor de la lista	. SELECT LEAST (NOTA1, NOTA2, NOTA3) FROM NOTAS
Funciones de cadenas de caracteres		
. CHR (n)	Devuelve el carácter de código ASCII n	. SELECT CHR (65) FROM DUAL A
. ASCII (cad)	Devuelve el código ASCII de la primera letra de 'cad'	. SELECT SCII ('Andoni') FROM DUAL 65
. CONCAT (cad1, cad2) ó cad' cad'	Concatena 'cad1' con 'cad2'	. SELECT CONCAT (' Soy ',NOM) FROM EMPLE . SELECT ' Soy ' NOM FROM EMPLE Soy LANDER
. LOWER (cad)	Devuelve 'cad' todo en minúsculas	. SELECT LOWER (NOM) FROM EMPLE lander
. UPPER (cad)	Devuelve 'cad' todo en mayúsculas	. SELECT UPPER (NOM) FROM EMPLE LANDER
. INITCAP (cad)	Devuelve el 1º carácter de cad en mayúsculas y el resto en minúsculas	. SELECT INITCAP (NOM) FROM EMPLE Lander
. LPAD (cad1, n [, cad2])	Añade a la izquierda de 'cad1' los caracteres de 'cad2' hasta tener una cadena de longitud n	. SELECT LPAD ('Lander', 8, '-') FROM DUAL --Lander . SELECT LPAD ('Lander', 8) FROM DUAL ' Lander'
. RPAD (cad1, n [, cad2])	Añade a la derecha de 'cad1' los caracteres de 'cad2' hasta tener una cadena de longitud n	. SELECT RPAD ('Lander', 8, '-') FROM DUAL Lander-- . SELECT RPAD ('Lander', 8) FROM DUAL 'Lander '
. LTRIM (cad [, set])	Devuelve 'cad' con el grupo de caracteres de 'set' omitidos por la izquierda	. SELECT LTRIM ('abaLLL', 'a') FROM DUAL bLLL . SELECT LTRIM (' abaLLL') FROM DUAL 'abaLLL'
. RTRIM (cad [, set])	Devuelve 'cad' con el grupo de caracteres	. SELECT RTRIM ('LLL;::;', ';') FROM DUAL LLL

	de 'set' omitidos por la derecha	. SELECT RTRIM (' abaLLL ') FROM DUAL	' abaLLL'
. REPLACE (cad, cad1 [, cad2])	Devuelve 'cad' con cada ocurrencia 'cad1' sustituida por 'cad2'	. REPLACE ('abcdabe','ab','xx') FROM DUAL . REPLACE ('abcdabe','b') FROM DUAL	xxcdxxe 'a cda e'
. SUBSTR (cad, inicio [,n])	Devuelve el trozo de 'cad' que empieza en la posición <i>inicio</i> y tiene de longitud n	. SUBSTR ('ABCDEFG', 3, 2) FROM DUAL . SUBSTR ('ABCDEFG', -3, 2) FROM DUAL . SUBSTR ('ABCDEFG', 3) FROM DUAL	CD EF CDEFG
. TRANSLATE (cad1, cad2, cad3)	Devuelve 'cad1' con los caracteres encontrados en 'cad2' sustituidos por los caracteres de 'cad3'	. TRANSLATE ('abcbaf', 'ab', '12') . TRANSLATE ('abcbaf', 'ab')	12c21f 'cf'
. INSTR (cad1, cad2 [,comienzo [,m]])	Devuelve la posición de la m-ésima ocurrencia de 'cad2' en 'cad1' empezando la búsqueda en la posición <i>comienzo</i>	. INSTR ('Guadalupe', 'a', 2, 2) . INSTR ('Guadalupe', 'a', -5, 2) . INSTR ('Guadalupe', 'u')	5 3 2
. LENGTH (cad)	Devuelve el número de caracteres de 'cad'	. LENGTH ('Urritxe')	7
Funciones para manejo de fechas			
. SYSDATE	Devuelve la fecha del sistema	. SELECT SYSDATE FROM DUAL	02/10/02
. ADD_MONTHS (fecha, n)	Añade n meses a la fecha	. SELECT ADD_MONTHS ('17/01/02', 2) FROM DUAL	17/03/02
. LAST_DAY (fecha)	Devuelve la fecha del último día de ese mes	. SELECT LAST_DAY ('01/03/02') FROM DUAL	31/03/02
. MONTHS_BETWEEN (fecha1, fecha2)	Devuelve a diferencia en meses entre ambas fechas	. SELECT MONTHS_BETWEEN (SYSDATE, fechaNac)/12	Calcularía la edad
. NEXT_DAY (fecha, cad)	Devuelve la fecha del primer día de la semana indicado por 'cad' después de la fecha indicada por <i>fecha</i>	. SELECT NEXT_DAY (SYSDATE, 'JUEVES')	03/10/02

Funciones de conversión		
. TO_CHAR (FECHA, 'formato')	Convierte la fecha de tipo DATE a tipo VARCHAR2 con el formato especificado	<p>Si SYSDATE=17/10/02, haciendo SELECT TO_CHAR...</p> <pre> ... (SYSDATE, 'yyyy') FROM DUAL 2002 ... (SYSDATE, 'yy') FROM DUAL 02 ... (SYSDATE, 'q') FROM DUAL 3 ... (SYSDATE, 'mm') FROM DUAL 10 ... (SYSDATE, 'month') FROM DUAL OCTUBRE ... (SYSDATE, 'q') FROM DUAL 3 ... (SYSDATE, 'dd') FROM DUAL 17 ... (SYSDATE, 'mon') FROM DUAL OCT </pre>
. TO_CHAR (número, 'formato')	<p>Convierte un número de tipo NUMBER a tipo VARCHAR2 con el formato especificado</p> <p>Máscaras:</p> <p>9 → Devuelve el valor, sin ceros a la izquierda</p> <p>0 → Muestra 0 si es 0, con ceros al principio</p> <p>\$ → Muestra \$ a la izquierda del valor</p> <p>B → Muestra espacios si es 0</p> <p>MI → Si es un valor negativo, - sigue al número</p> <p>S → + si es positivo, - si es negativo</p> <p>PR → Si es negativo se muestra entre < ></p> <p>D → Carácter decimal en la posición de D</p>	<p>Más máscaras:</p> <p>G → Carácter de miles en la posición de G</p> <p>C → Devuelve el símbolo ISO del territorio</p> <p>L → Devuelve el símbolo de la moneda</p> <p>, → Devuelve la coma en esa posición</p> <p>. → Devuelve el punto en esa posición</p> <p>V → Devuelve el valor multiplicado por 10ⁿ, donde n es el número de nueves después de la 'V'</p> <p>EEEE → Devuelve el valor usando notación científica</p> <p>RN → Devuelve el valor en números romanos</p> <p>FM → Devuelve el valor alineado a la izquierda</p>
. TO_NUMBER (cad [, 'formato'])	Convierte 'cad' a tipo NUMBER con el formato especificado. No puede haber espacios entre números ni otros caracteres, excepto el carácter decimal o el signo menos a la izquierda	. SELECT TO_NUMBER ('123,99', '999d99') FROM DUAL 123,99
. TO_DATE (cad, 'formato')	Convierte 'cad' de tipo VARCHAR2 o CHAR, a un valor de tipo DATE con el formato especificado	. SELECT TO_DATE ('01012001', 'DD/MM/YYYY') 01/01/2001

Funciones e instrucciones de administración de Oracle

. CREATE USER usu1 IDENTIFIED BY password [DEFAULT TABLESPACE tablespace] [TEMPORARY TABLESPACE tablespace] [QUOTA {entero {K M} UNLIMITED} ON tablespace] [PROFILE perfil];	Creación del usuario USU1 con contraseña de acceso PASSWORD. TABLESPACEs por defecto y temporal. Tamaño de cada tablespace en Kbytes o Mbytes. Puede ser ilimitado. Asignación de un perfil para USU1.	. CREATE USER josu IDENTIFIED BY Josu DEFAULT TABLESPACE trabajo QUOTA 550K ON trabajo TEMPORARY TABLESPACE trabajo
. ALTER USER usu1 IDENTIFIED BY password [DEFAULT TABLESPACE tablespace] [TEMPORARY TABLESPACE tablespace] [QUOTA {entero {K M} UNLIMITED} ON tablespace] [PROFILE perfil];	Modificación de la definición del usuario USU1.	
. DROP USER usu1 [CASCADE]	Eliminación del usuario USU1. La opción <i>CASCADE</i> suprime todos los objetos del usuario antes de borrarlo	
. GRANT {priv_obj1 [,priv_obj2]... ALL} [(col1 [, col2]...)] ON [usuario.]objeto TO {usuario1 rol1 PUBLIC} [,{usuario2 rol2 PUBLIC}...] [WITH GRANT OPTION] ;	Adjudica privilegios sobre <u>objetos</u> o <u>columnas de objetos</u> a un usuario o rol. PUBLIC adjudica los privilegios a todos los usuarios actuales o futuros. WITH GRANT OPTION permite que el receptor del privilegio o rol se lo asigne a otros usuarios o roles.	
. GRANT {priv1 rol1} [{priv2 rol2},...]	Adjudica privilegios de <u>sistema</u> a un usuario	

TO {usuario1 rol1 PUBLIC} [{usuario2 rol2 PUBLIC}]... [WITH ADMIN OPTION] ;	o rol. PUBLIC adjudica los privilegios a todos los usuarios actuales o futuros. WITH ADMIN OPTION permite que el receptor del privilegio o rol pueda asignar esos mismos privilegios de administrador a otros usuarios o roles.	
. REVOKE {priv_obj1 [,priv_obj2]... ALL} ON [usuario.]objeto FROM {usuario1 rol1 PUBLIC} [{usuario2 rol2 PUBLIC}]...;	Retira privilegios sobre <u>objetos</u> o <u>columnas de objetos</u> a un usuario o rol. PUBLIC retira los privilegios a todos los usuarios actuales o futuros.	
. REVOKE {priv1 rol1} [{,priv2 rol2}]... FROM {usuario1 rol1 PUBLIC} [{usuario2 rol2 PUBLIC}]...;	Retira privilegios de <u>sistema</u> a un usuario o rol. PUBLIC retira los privilegios a todos los usuarios actuales o futuros.	
. CREATE ROLE rol [IDENTIFIED BY password];	Crea el rol ROL con, opcionalmente, contraseña PASSWORD. A este rol habrá que adjudicarle privilegios con la instrucción GRANT	
. DROP ROLE rol;	Elimina el rol ROL.	
. ALTER USER usu1 DEFAULT ROLE rol;	Adjudicación de un rol por defecto al usuario USU1	

<pre> . CREATE PROFILE perfil1 LIMIT {SESSIONS_PER_USER CPU_PER_SESSION CPU_PER_CALL CONNECT_TIME IDLE_TIME LOGICAL_READS_PER_SESSION LOGICAL_READS_PER_CALL PRIVATE_SGA COMPOSITE_LIMIT } {Entero {K M} UNLIMITED DEFAULT} [{SESSIONS_PER_USER CPU_PER_SESSION CPU_PER_CALL CONNECT_TIME IDLE_TIME LOGICAL_READS_PER_SESSION LOGICAL_READS_PER_CALL PRIVATE_SGA COMPOSITE_LIMIT } {Entero {K M} UNLIMITED DEFAULT}]... ; </pre>	<p>Creación del perfil PERFIL1 con las limitaciones indicadas.</p>	
<pre> . DROP PROFILE perfil1 [CASCADE]; </pre>	<p>Borra el perfil PERFIL1. <i>CASCADE</i> borrará los usuarios con ese perfil</p>	

<pre> CREATE TABLESPACE tablespace1 DATAFILE 'arch1' [SIZE entero [K M] [REUSE] [, 'arch2' [SIZE entero [K M] [REUSE]... [DEFAULT STORAGE (INITIAL tamaño NEXT tamaño MINEXTENTS tamaño MEXEXTENTS tamaño PCTINCREASE valor)] [ONLINE OFFLINE] ; </pre>	<p>Creación del tablespace TABLESPACE1</p>	
<pre> ALTER TABLESPACE tablespace1 { [ADD DATAFILE 'arch1' [SIZE entero [K M] [REUSE] [AUTOEXTEND ON OFF] [, 'arch2' [SIZE entero [K M] [REUSE] [AUTOEXTEND ON OFF]...] [RENAME DATAFILE 'arch1' [, 'arch2']... TO 'archi1' [, 'archi2']...] [DEFAULT STORAGE clausAlmacenamto] [ONLINE OFFLINE] }; </pre>	<p>Modificación de la tablespace.</p>	
<pre> DROP TABLESPACE tbalespace1 [INCLUDING CONTENTS] ; </pre>	<p>Elimina el tablespace. INCLUDING CONTENTS permite borrar un tablespace que tenga datos.</p>	